

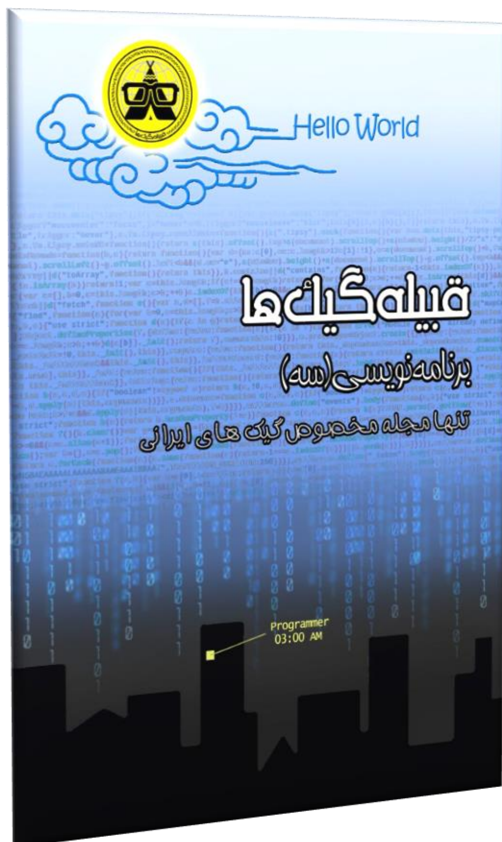
Hello World

قبیله گیک ها

برنامه نویسی (سه)

تنها مجله مخصوص گیک های ایرانی

Programmer
03:00 AM



قبیـــــــــله گیـــــــــک ها



ماهنامه

علمی – تخصصی

کامپیوتری

سال اول

شماره نه

مرداد ۹۵

سردبیر: @BoBzBoBz

- طراح گرافیک: @An0nym0u3

[کانال ما در تلگرام]

[گروه ارتباط با اعضای قبیله گیک ها]

گروه نویسندگان:

- @abdolhayb
- @ An0nym0u3
- @BoBzBoBz
- @Badrinex
- @Geek_072
- @mansourebrahimi
- @Mehnaty
- @rooham_inet
- @Sh_Ebrahiimii
- @THEnoneIDENtity

فهرست



سخن سردبیر

با سلام خدمت شما همراهان گرامی...

بار دیگر با شماره جدیدی از مجله قبیله گیک ها در خدمت شما هستیم.

این شماره نهمین شماره مجله قبیله گیک ها و سومین شماره پیاپی در زمینه برنامه نویسی میباشد.

ما متوجه شدیم که تمرکز شدید بر روی یک موضوع باعث خسته شدن شما خوانندگان عزیز میشود و بهتر است با تنوع موضوعات مختلف در هر شماره طیف وسیعتری از موضوعات و بطبع آن گیک هایی بیشتری با سلایق مختلف تر را مخاطب خود قرار دهیم، بنابراین تصمیم گرفتیم در ۳ شماره آینده قبیله گیک ها موضوع مجله را باز بگذاریم و هر کدام از نویسندگان میتوانند در مورد هر موضوعی که تمایل دارند بنویسند تا هم مطالب متفاوت در مجله داشته باشیم و هم روحی دیگر به مجله و خوانندگان آن تزریق شود.

لطفا نظرات خود را در این مورد با ما در میان بگذارید.

ما در ماه گذشته رنگ لوگوی گروه چت قبیله گیک ها را در تلگرام به آبی تغییر دادیم تا شما بتوانید به راحتی بین کانال اصلی با لوگوی سیاه و گروه قبیله گیک ها با لوگوی آبی تفاوت قایل شوید.

خبر دیگری که خوشحالیم با شما در میان بگذاریم تولد اپلیکشن "قبیله گیک ها" برای سیستم عامل اندروید میباشد.

شما با استفاده از این اپلیکشن میتوانید بهتر با قبیله و اعضای آن در ارتباط باشید و همچنین میتوانید به آخرین اخبار قبیله گیک ها دسترسی داشته باشید. این اپلیکشن در حال گسترش میباشد و ما قصد داریم در آینده آموزشهایی را از این طریق با اعضای قبیله به اشتراک بگذاریم .

برای دریافت این اپلیکشن به کانال قبیله گیک ها مراجعه نمایید.

دقت نمایید که برای اطمینان این اپلیکشن را فقط از طریق کانال رسمی قبیله گیک ها دریافت نمایید.

دوستداران استیکرهای قبیله گیک ها میتوانند این ماه از ۲ استیکر جدید که به بسته استیکرهای قبیله گیک ها اضافه شده است استفاده نمایند.

همچنین تیم طراحی استیکر قبیله گیک ها از پیشنهادات و طرح های شما در زمینه استیکر صمیمانه استقبال میکند.

دوستانی که در کانال قبیله گیک ها عضو هستند حتما در جریان انتشار پوسترهایی با موضوع قبیله گیک ها شده اند. این پوسترها به نوعی حال و هوای تیم نویسندگان قبیله گیک ها در ماه گذشته را در قالب تضاد بین سنت و مدرنیت نشان میدهد که البته در آینده و با توجه به باز بودن موضوع مجله در شماره های آینده میتوانیم منتظر پوسترهای جدید با موضوعات متفاوت تری باشیم.

در پایان مانند همیشه از شما درخواست میکنیم که هرگونه پیشنهاد، انتقاد و یا نظر خود در زمینه قبیله گیک ها را با ما در میان بگذارید.

برای ارتباط با ما میتوانید از لینک زیر استفاده نمایید:

https://telegram.me/joinchat/BMcoYj4zho8sEcN-QP_iUg

@BoBzBoBz

Tehran Institute of Technology

مجتمع فنی تهران نمایندگی میرداماد

صفر تا صد طراحی وب و برنامه نویسی را با ما باشید

HTML

گواهینامه معتبر

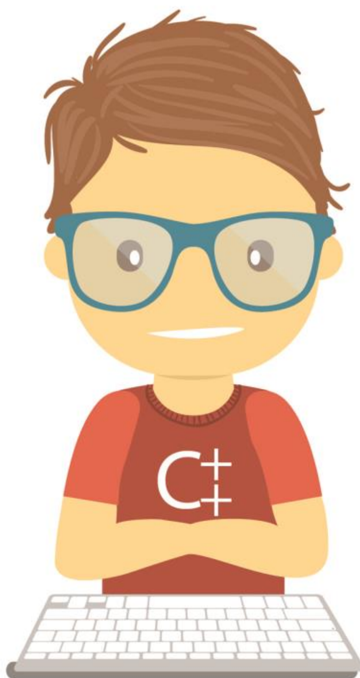


PHP



حرفه‌ای ترین اساتید





AJAX

CSS





کلاس‌های کاملا کاربردی
و پروژه محور

۰۲۱ – ۲۲ ۲۲ ۲۸ ۱۶

www.mftmirdamad.com

it@mftmirdamad.com

Java laravel perl Programming Language php python Ruby ANGULARJS Microsoft® .NET ZEND FRAMEWORK



قدرت خاص

برنامه نویسی برای هر دیوایسی نیاز به ابزار و نرم افزار های مرتبط با آن را دارد تا برنامه نویسان بتوانند به صورت مستقیم برنامه را برای دیوایس طراحی کنند . شرکت های نظیر بلک بری و اپل چون از سیستم عامل مختص به خود استفاده می کنند باید نرم افزاری برای ساخت برنامه داشته باشند. به طور مثال اپل از نرم افزار **Xcode** برای ساخت اپلیکیشن دیوایس های خود استفاده می کند و این نرم افزار را به صورت رایگان می توان از **App Store** سیستم عامل مکینتاش دانلود کرد.



تاریخچه xcode:

وقتی اپل محصولات خود را معرفی کرد ،نیاز به یک ابزار برای ساخت برنامه مرتبط با دیوایس ها به شدت احساس می شد، بنابراین تصمیم به ساخت نرم افزاری گرفتند که بتواند این نیاز را رفع کند. اولین نسخه **xCode** در پاییز سال ۲۰۰۳ به منظور انجام ساخت پروژه ارائه شد که دارای محیط گرافیکی بوده است.این نسخه اولیه باگ های داشت که با بروزرسانی به نسخه ۱،۵ آن مشکلات رفع شد.

نسخه بعدی ارائه شده نرم افزار **xcode** برای سیستم عامل مکینتاش ۱۰،۴ یا **tiger** ارائه شد. این نسخه دارای ویژگی **Quartz Composer** باشد که باعث شده است حس برنامه نویسی برای جاوا نمایش داده شود همچنین رفرنسی برای چگونگی نصب در سیستم وآموزش کار با این این نرم افزار به صورت داکیومنت در سایت اپل موجود می باشد .



از سال ۲۰۰۳ تا کنون که حدود ۱۳ سال از اولین ارائه نرم افزار **xcode** می گذرد می توانید در مقایسه با نسخه های که در سالهای بعد منتشر شد به روند پیشرفت و ارتقا قابلیت های برنامه اشاره کرد که در گذر زمان چه امکاناتی به این نرم افزار افزوده شده است . برای درک بهتر تغییرات اعمال شده می توانید در اینترنت نام **xcode** را جستجو کنید.

امکانات xcode:

نرم افزار **xcode** از قابلیت های بسیاری برخوردار می باشد که تعدادی از ویژگی های این ابزار به شرح ذیل است:

Source Editr:

دارای ویرایشگر کد حرفه ای می باشد و به صورت پیشرفته کد را پردازش و به پایان می رساند،اگر در کد ها پیغام خطای وجود داشته باشد از لحاظ **syntax** آن را های لایت می کند.

Interface Builder Built In:

طراحی و آزمایش رابط کاربری بدون استفاده از یک خط کد، ایجاد نمونه اولیه در کمترین زمان و در نهایت نمای گرافیکی را با رابط بوسیله ویرایشگر **xcode** به منابع متصل می کند.

iOS Simulator:

به وسیله **iOS SDK** و **xCode** می توان برنامه ی مورد نظر را طراحی و سپس آن را نصب و اجرا کرد. شبیه ساز **iOS** در سیستم عامل **Mac OS X** وجود دارد که قادر است برنامه ساخته شده را نمایش دهد تا از سالم بودن و درست کار کردن اپلیکیشن به صورت دقیق اطمینان کامل را داشته باشد.

Zombie Detection:

برای پیدا کردن پیغام خطا و عوامل خرابی در برنامه با ایجاد یک تله زمانی که برنامه سعی می کند برای خود دسترسی ایجاد کند اما عدم حافظه کافی برای اینکار مانع از این کار می شود.

Script languages:

سیستم عامل مکینتاش با زبان های برنامه نویسی **AppleScript**، **Perl, Python** و **Ruby** که به صورت آماده به نصب می باشد بوجود آمده زیرا سیستم عامل مکینتاش خود از **unix** ساخته شده است. پس بنابراین شما می توانید به راحتی با هر نسخه از زبان مورد علاقه خود که **open source** می باشد با نرم افزار **xcode** یا ابزار های مشابه این که در بسته های سیستم عامل وحود دارد اپلیکیشن خود را بسازید.

Storyboards:

یکبرنامه **iOS** کامل می باشد که از دیدگاه های متفاوت کاربران جهت پیشرفت در انجام پروژه تشکیل شده است. این دیدگاه ها رابطه ای مستقیم با تعریف **Storyboards** دارد. به طوری که چارت یک برنامه را می توان رصد کرد.سازنده رابط کاربری **storyboard** به آسانی می تواند نمایه های جدید طراحی و ایجاد کند. در نهایت آنها را با هم ادغام کرده تا یک رابط کاربری کامل

برای کد نویسی حاضر و آماده باشد.



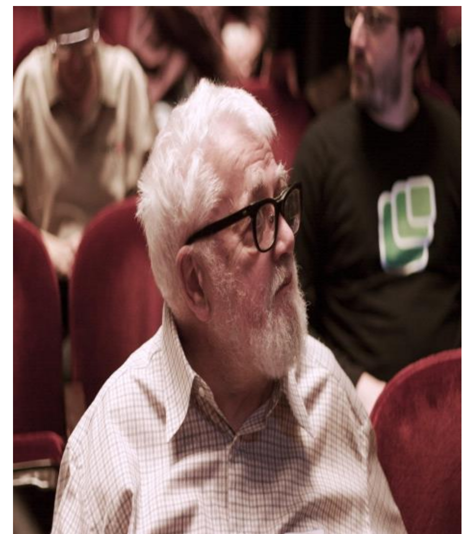
کسب درامد:

اپل برای برنامه نویسان این امکان را براورده کرده تا هر برنامه نویس بتواند برنامه خود را بسازد و در app store برای فروش قرار دهد.برای اینکار قرار دادی با اپل بسته می شود و شما در سودی که از فروش اپلیکیشن به دست می آورید درصدی از آن را با اپل شریک خواهید شد.

نرم افزار xcode به صورت انحصاری برای شرکت اپل است. اگر کسی بخواهد با این نرم افزار برنامه ای برای دیوایس های اپل بنویسد و طراحی کند می بایست از سیستم عامل مکینتاش برخوردار باشد تا این عمل انجام پذیرد. از معایب این برنامه می توان به انحصاری بودن آن اشاره کرد ، ولی دارای مزایای زیادی می باشد به طور مثال محیط زیبا و ساده ای دارد ، اکثر زبان های برنامه نویسی را پشتیبانی می کند و به صورت کاملا و صد درصد رایگان می باشد و از تمامی ابزار های که در این نرم افزار قرار داده شده است می توان استفاده کرد.



زبان lisp در اواخر دهه ۵۰ توسط John McCarthy در MIT بوجود آمد و از قدیمی ترین زبانهای برنامه سازی است و هدف عمده آن انجام محاسبات نمادین است. مک کارتی با همکاری کلود شانون و ماروین مینسکی یک کارگاه آموزشی با موضوع هوش مصنوعی برگزار کرد و پس از آنکه موضوع هوش مصنوعی به طور جدی مطرح گردید مک کارتی کار روی بازی های هوشمندانه ماشینی را آغاز کرد و ازحاصل این کار زبان Lisp پدیدار گردید، او این زبان را برای پردازش لیست ها ابداع کرد. البته هم اکنون از این زبان به شکل گسترده‌ای در حوزه هوش مصنوعی استفاده می‌شود.



این زبان به جای آنکه از منطق ریاضی و کار روی اعداد استفاده کند،علایم و نماد ها را به اشیا تغییر می دهد. یعنی از تعدادی لیست برای توصیف منطق کاری برنامه بهره می برد و در نهایت، خروجی این زبان تعدادی جمله یا عبارت توصیفی خواهد بود.

Example LISP Code

```
(define (reduce f a x y b fx fy)
  (cond ((close-enough? a b) x)
        ((> fx fy)
         (let ((new (x-point a y)))
           (reduce f a new x y (f new) fx)))
        (else
         (let ((new (y-point x b)))
           (reduce f x y new b fy (f new))))))
```

SICP, Technical Report 735, 1983

البته امروزه از شکل های تازه تری از زبان Lisp در سیستم های خبره یا Expert و برنامه های پردازش زبان طبیعیNLP استفاده می شود. در ضمن این زبان به قدری سطح بالابود که تازه در اواخر دهه ۸۰ میلادی کامپیوترهایی پدید آمدند که توان کامل پردازش دستورهای این زبان را داشتند. همه کسانی که نخستین گام ها را در راه معرفی هوش مصنوعی برداشتند، یک هدف را در سر داشتند و آن رساندن سطح هوش ماشینی به سطح هوش انسانی بود .

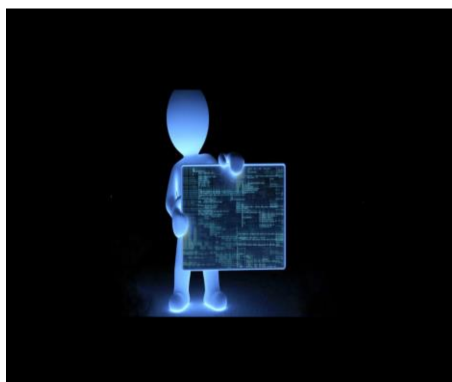
ویژگی ها:

از نظر تئوری می توان یک کامپیوتر سخت افزار یا میان افزار ایجاد کرد تا برنامه هایی را که به یک زبان برنامه سازی خاص نوشته شده اند مستقیما اجرا کنند،بدین ترتیب یک کامپیوتر Lisp ساخته می شود.Lisp معمولا با مفسر نرم افزاری پیاده سازی می شود و در اینگونه پیاده سازی زبان مترجم کد ماشین را برای کامپیوتر تولید نمی کند. زبان Lisp از جدول نماد در زمان اجرا استفاده می کند. زبان Lisp از قوائد ایستا استفاده نمی کند و در نتیجه هر ارجاع به نامی در این زبان ، مستلزم فرآیند تفسیر پیچیدگی و گرانی است.

در Lisp در هر نقطه ای از برنامه می توان عنصر جدیدی را به ساختار آن اضافه کرد و حافظه می تواند در هر نقطه ای از برنامه آزاد شود. این زبان از دسته زبان هایی است که برای استنتاج حقایق در مورد داده ها بکار می رود.

نسخه های مختلف Lisp

اولین نگارش آن که شامل یک مفسر، یک کامپایلر و مکانیزم تخصیص حافظه و برگشت حافظه پویا بود،ایجاد شد. یک سال بعد اولین زبان استاندارد با نام LISPY.S معرفی شد، پس از آن تعدادی از نسخه‌ها و محیط‌های برنامه‌نویسی Lisp توسعه یافتند، مانند MacLisp و FranzLisp و InterLisp. . هرچند در بعضی از جزئیات خاص متفاوتند، ولی هسته اصلی ساختار Syntax و ارتباط بین آنها اساسا یکسان است و پر استفاده‌ترین نسخه‌های این زبان ، Common Lisp و Schema هستند. در کنار Lisp تعدادی از زبان‌های برنامه‌نویسی تابعی دیگر نیز توسعه یافتند، برای مثال ML که برگرفته از Meta Language است که این زبان یک زبان برنامه‌نویسی تابعی با دامنه ایستا است و تفاوت اصلی آن با زبان Lisp در Syntax آن است و بیشتر شبیه زبان برنامه‌نویسی پاسکال است.



و در آخر اینکه

Lisp به برنامه نویس قدرت کامل برای اتصال به ساختارهای اطلاعاتی را می دهد.این زبان یکی از قدیمی ترین زبانهای محاسباتی است که دقت کافی در برنامه نویسی و طراحی توسعه باعث شده است هنوز فعال باقی بماند.

بزودی برنامه نویسی با نمادهایی که نیاز به سطح بسیار بالا تری از چکیدگی دارند تولید می شوند، حتی پیشرفته تر از امکاناتی که با زبانهای برنامه نویسی مخصوص پردازش اعداد ممکن بود مانند زبان برنامه نویسی فرترن.



@mehnaty

Shell Programming

در قسمت اول این مقاله به بررسی برنامه نویسی در محیط Shell پرداختیم و همچنین برخی از دستورات اساسی در محیط Shell را بررسی کردیم. حال به ادامه این مقاله می پردازیم.

شرط ها و دستورات شرطی:

یکی از اساسی ترین ویژگی هایی که هر زبان برنامه نویسی باید دارا باشد قابلیت تست کردن شرط ها و کار کردن با دستورات شرطی است. مسلماً پوسته نیز از این قاعده جدا نیست. برای انجام اعمال شرطی در پوسته باید از دستور test استفاده نمایید. البته این دستور معمولا به عنوان بخشی از (if) به کار می رود و به همین دلیل من در اینجا این دستور را به همراه (if) به شما معرفی کرده ام. البته شایان ذکر است که می توانید این دستور را به همراه بسیاری از دستورات دیگر ترکیب نمائید و کاربرد آن فقط به دستور (if) محدود نمی گردد. شکل کلی این دستور به شکل زیر است:

```
If test-option something
Then
Statement
fi
```

در دستور بالا پس از دستور (if) دستور تست آمده است که به مفسر پوسته می گوید می خواهیم از این دستور استفاده نمائیم. پس از آن (-option) آمده است که یکی از سوئیچ های دستور (test) است که برای انجام و چک کردن عمل خاصی استفاده می شود و بر حسب صحیح یا غیر صحیح بودن شرط یکی از مقادیر (true) یا (false) را بر می گرداند در قسمت (something) هم باید عبارتی را بنویسید که می خواهید مورد بررسی قرار بگیرد. باید در نظر داشته باشید در اینجا شکل کامل دستور (if) ذکر نشده است. شکل کامل این دستور را می توانید در صفحات بعدی مشاهده نمائید.در قسمت (statement) می توانید دستوراتی را که می خواهید اجرا شوند، بنویسید.

در جداول زیر لیستی از آپشن هایی که می توانید در دستور test مورد استفاده قرار دهید آمده است. تنها نکته باقیمانده این است که می توانید به جای نوشتن دستور (test) شرط ها و عبارات خود را بین[ ] نیز قرار دهید و از نوشتن (test) صرف نظر نمائید. این کار ظاهر خوشایندتری به دستور شما خواهد داد.

برای تست مقادیر رشته ای در دستور می توانید از جدول زیر استفاده نمایید:

توضیح	عملگر مقایسه ای
نتیجه ی عبارت زمانی صحیح است که هر دو رشته با یکدیگر برابر باشند.	String1=string2
نتیجه ی عبارت زمانی صحیح است که دو رشته با یکدیگرمخالف باشند	String1!=string2
در صورتی صحیح است که رشته خالی نباشد	-n string
در صورتی صحیح است که رشته خالی باشد	-z string

برای مقایسه عبارتهای صحیح می توانید از جدول زیر استفاده نمایید

توضیح	سوییچ ها
زمانی درست است که دو عبارت مساوی باشند	Exp1 –eq exp2
زمانی درست است که دو عبارت مساوی نباشند	Exp1 –ne exp2
زمانی درست است که عبارت اول بزرگتر از عبارت دومی باشد	Exp1 –gt exp2
زمانی درست است که عبارت اول بزرگتر یا مساوی عبارت دومی باشد	Exp1 –ge exp2
زمانی درست است که عبارت اول کوچکتر از عبارت دومی باشد	Exp1 –it exp2
زمانی درست است که عبارت اول کوچکتر یا مساوی عبارت دومی باشد	Exp1 –le exp2
همان not یا نقیض است	! expression

برای انجام اعمال شرطی روی فایل ها از گزینه های جدول زیر استفاده نمایید

توضیحات	Options
در صورتی صحیح است که فایل یک دایرکتوری باشد	-d
در صورتی صحیح است که فایل وجود داشته باشد	-f
در صورتی صحیح است که سایز فایل بزرگتر از صفر باشد	-s
در صورتی صحیح است که فایل قابل خواندن باشد	-r
در صورتی صحیح است که فایل قابل نوشتن باشد	-w
در صورتی صحیح است که فایل قابل اجرا باشد	-x

ساختار های کنترلی و شرطی

همانطور که تا اینجا متوجه شده اید، پوسته یکسری دستورات شرطی و کنترلی دارد که در بخش قبلی به طور مختصر با یکی از آنها یعنی (if) آشنا شدید. در این بخش قصد دارم شما را به طور کامل با این ساختار آشنا کنم.

اولین ساختاری که با هم در اینجا بررسی می کنیم ساختار (if) است. همانطوری که می دانید از این ساختار برای بررسی یک شرط استفاده می شود و در صورت درست بودن شرط یکسری دستورات و در صورت لزوم و یا در صورت ناصحیح بودن شرط، یکسری دستورات دیگر اجرا می شوند. البته می توان این دستور را به صورت تو در تو هم استفاده کرد .شکل کلی این دستور به شکل زیر است:

صورت [=”yes”] در خواهد آمد. اما اگر این دو مقدار بین دو "" قرار گیرد شرط به صورت [“”=”yes”]تلقی شده و false نتیجه می دهد.

نکته: دستور echo به صورت پیش فرض پس از چاپ دستور به خط بعدی می رود اگر می خواهید این عمل

```
Echo –n “sometext”If condition;then
True body
```

شکل دستوری و تو در توی (if) به شکل زیر است:

```
If condition; then
True body
Elif condition;then
Statement
Elif condition;then
.
.
.
Else
Falsebody
Fi
```

همانطوری که مشاهده می کنید این ساختار ها از نظر ظاهری شباهت بسیار زیادی به ساختار های شرطی زبان های دیگر دارد. خوب به مثال زیر توجه کنید:

```
#!/bin/sh
Echo"is it morning?(please answer yes or no)"
Read timeofday
If["$timeofday"="yes"]
Then
Echo"good morning"
Elif["$timeofday"="no"]
Then
Echo"good afternoon"
Else
Echo "sorry $timeofday not recognized.enter yes or no"
Fi
```

همانطور که می بینید استفاده از ساختار (if) به همراه دستور(test) بسیار ساده است. تنها نکته ای که در این مثال وجود دارد این است که در ساختار (test) از"$timeofday" به جای $timeofday استفاده شده است. این نوع ساختار بسیار متداول است و به این دلیل به کار می رود که اگر کاربر هیچ مقداری را در ورودی وارد نکند و فقط enter را فشار دهد، اگر فقط $timeofday نوشته شده باشد خطا ایجاد خواهد شد چرا که شرط به انجام نشود می توانید از سوییچ n– دستور echo به صورت زیر استفاده نمایید:

```
Else
False body
Fi
```

شکل کلی این دستور بسیار ساده است. تنها نکته ای که باید به آن توجه نمائید این است که اگر می خواهید (then) را در همان خطی که (if) را نوشته اید بنویسید باید قبل از آن (;) را قرار دهید در غیر این صورت نیازی به (;) نیست.

دستور for : ما از این دستور برای گردش بین یکسری از مقادیر استفاده می نمائیم. بر خلاف حلقه for بین زبان های برنامه نویسی مختلف کاربرد این حلقه در shell متفاوت است و کاربرد آن بیشتر شبیه به -for each است تا for عادی در هر صورت شکل کلی دستور زیر به شکل زیر است:

```
For variable in value
Do
   Statement
Done
```

به مثالی در مورد حلقه for توجه نمایید:

```
For foo in bar fud 43
Do
Echo $foo
Done
```

خروجی این برنامه به شکل زیر است:

```
Bar
Fud
43
```

امیدوارم این مقاله برای شما گیک های عزیز مفید بوده باشد.

محمد مهدی خلعتبری
@Geek_072



The World is "Open Source"



Message Box

MOM:
Dinner

Auto Reply:
OK





Command Center

Coffee

0%

Program

100%

Sleep

75%

Internet

Connected

PROGRAMMER UNIVERSE

Summary:
3269 line


Alerts:

Need more COFFEE

@An0nym0u3

Ruby

اگر با زبان های **perl** و **python** کار کردید و از هر دوی آن ها خوشتان آمده و به دنبال زبانی هستید که هم مانند **python** شی گرا و هم به اندازه **perl** قدرتمند باشد و زبانی که مانند **smaltalk** / **eiffel** و **lisp** و مخلوطی از آن ها باشد؛نام این زبان **Ruby** است.



Ruby یک زبان برنامه نویسی مفسری / متن باز و اسکریپت نویسی است که شیوه های مختلف برنامه نویسیِ شی گرایی / تابعی و باز تابی را شامل می شود.طراحی این زبان بسیار ساده است ولی در عین حال بسیار قدرتمند و کامل می باشد و توانایی توسعه پذیری زیادی دارد(بیشتر بر روی سیستم عامل لینوکس توسعه یافته است).**Ruby** با زبان **c** توسط یوکیهیر ماتسوموتو ژاپنی ساخته شده است.ماتسوموتو در ساخت آن از زبانی هایی که خودش از آن ها استفاده می کرد کمک گرفته است(**lisp/eiffel/smaltalk**).مفسر **Ruby** تحت اجازه نامه همگانی گنو عرضه می شود ولی حق کپی رایت دارد.این زبان در سال ۲۰۰۶ با استقبال زیادی مواجه شد و هم اکنون جزو ده زبان برتر برنامه نویسیِ سایت **TIOBE.COM** قرار دارد(از رتبه ۱۶ به رتبه ۱۰ آمده است.

بیشترین کاربرد **Ruby** در طراحی وب است و همچنین بیشترین افرادی که از **Ruby** استفاده می کنند طراحان وب می باشند. این زبان همچنین در شبکه / علوم مخابرات / برنامه نویسی ربات و... استفاده می شود.توسعه دهندگان بازی های ویدیویی گروهی دیگر از برنامه نویسان هستند که از این زبان استفاده می کنند همچنین مدیران سیستم نیز از این زبان استفاده می کنند و **Ruby** زبان محبوبی بین آن ها می باشد. اگر بخواهیم قدرت **Ruby** را به رخ دیگران بگشیم باید گفت که **twitter** با زبان روبی پیاده سازی شده و ابزار **chef** نیز با این زبان نوشته شده است که بیانگر قدرت این زبان است.



در مورد مزایای **Ruby** می توان گفت که این زبان شامل شیوه های مختلف برنامه نویسی و درآمیخته از چند زبان برنامه نویسی است.دارای رابط کاربری خوب و ساده و فریم ورک های بسیار قدرتمند می باشد.سرعت توسعه پذیری بالا/متوازن و دقیق بودن و غافل گیری کم از دیگر مزایای این زبان می باشند.کتابخانه های بسیار قدرتمندی دارد و کمپانی های زیادی از جمله **New railk** از این زبان پشتیبانی می کنند. مفسر **Ruby** همچنین بر روی اکثر سیستم عامل ها از جمله ویندوز / لینوکس/ مکینتاش/داس و... وجود دارد.

همچنین در مورد معایب **Ruby** می توان به اینکه یاد گیری روبی آسان نیست اشاره کرد **ruby** یک زبان جدید است و کد های منحصر به فرد خود را دارد به طوری که حتی برنامه نویسان ماهر نیز باید مدت زمان زیادی را صرف یاد گیری آن کنند.

در مقایسه با سایر زبان های برنامه نویسی (**php** و **C#**) کتاب ها و منابع آموزشی روبی بسیار کمتر است به طوری که از سال ۲۰۰۶ تا کنون تقریبا ۴۰۰ کتاب درمورد **C#** و ۵۰ کتاب **ruby** وجود دارد.سرعت پردازش ان در مقایسه با زبان های برنامه نویسی دیگر مانند **java** بسیار کمتر است.مدت زمان اپدیت های روبی خیلی دیر به دیر است.

در نهایت باید گفت زبان **Ruby**

برای تازه کاران که می خواهند برنامه نویسی را شروع کنند توصیه نمی شود پس روبی زبانی برای شروع برنامه نویسی نیست اما اگر برنامه نویسی نسبتا ماهری هستید و بهزبان های **Perl** و **Python** و **PHP** تسلط دارید و قصد یاد گیری یک زبان جدید و درآمدزا را دارید **Ruby** پیشنهاد می شود.

@THEnoneIDENtity

برنامه نویسی تحت شبکه یا ، برای شبکه؟

هنگامی که در ذهن خود به برنامه نویسی تحت شبکه فکر میکنید ، به یک دو راهی فنی بر میخورید ، آن هم درک کردن موضوع برنامه نویسی تحت شبکه است یا برای شبکه ؟

این دو گزینه با هم فرق دارند و دو راه جداگانه را در پیش میگیرند ، این مساله مهم است که در قدم اول باید فرق بین آن دو را دانست .

سوکت نویسی !

برنامه نویسی تحت شبکه : " معمولا این نوع برنامه نویسی و توسعه در سمت کنترل و دسترسی به IP و PORT انجام میشود که اکثر زبان های برنامه نویسی از آن پشتیبانی میکنند و هدف کلی آن ، ارسال دستور برای انجام فعالیت سرویسی در شبکه است . این نوع برنامه نویسی مشهور به سوکت نویسی یا socket programming است و محوری بر اسکریپتینگ روی شبکه دارد ."

برای مطالعه بیشتر به شماره هشتم مجله مراجعه کنید !

مثال :

```
<span style="font-family: tahoma,arial,helvetica,sans-serif; font-size: small;"><code>3481B "TCP over Second (2.5G) and Third (3G) Generation Wireless
</code></span>
```

در قطعه کد بالا مشاهده میکنید که با فراخوانی پورت سرویس مورد نظر ، برای آن فعالیتی در نظر میگیرد و دستور را به صورت یک کد استاندارد برای سرویس ارسال میکند .

سوکت نویسی بیشتر برای راه اندازی سرویس ها و کنترل شبکه کارایی دارد و لازمه شروع کار با آن ، شناخت و سواد مطلوب شبکه و امنیت است. از زبان های مشهور استفاده شده در این زمینه میتوان به #C و JAVA اشاره کرد .

اجرای مرکزی و پایگاه داده !

برنامه نویسی برای شبکه : " به صورت خلاصه ، برنامه نویسی برای شبکه به معنی ساختن برنامه و یا اجرای برنامه در شبکه و Domain به حساب می آید ، که این امر موجب میشود تا ما متوجه این موضوع شویم که ، در این نوع برنامه نویسی قرار نیست اسکریپت یا سوکت و حتی دستور سیستمی داشته باشیم ، بنابراین ، این نوع برنامه نویسی بیشتر کاربردی و سمت کاربر به حساب می آید . ولی فرایند اجرا و نحوه کار با آن در شبکه انجام میشود و دارای یک دیتا بیس مرکزی میباشد ."

اجرای مرکزی : " وقتی که یک برنامه کاربردی را در سرور به صورت مرکزی به اجرا در می‌آورید ، نیازمند اتصال کلاینت برای اجرای آن نرم افزار هستید . پس طرح این مساله بدین گونه است که ، ما یک شبکه عضو دامین داریم که کاربرانی مشخص دارد و سمت آن ها هم بنا بر موقعیت شغلی مشخص است ، پس برای اجرای برنامه نیاز به محدویت گذاری داریم و همچنین چند ارتباط Remote برای اتصال و استفاده از برنامه در سرور مورد نیاز میباشد ."

این روش در واقع برنامه نویسی نیست ! فقط نصب و اجرای یک نرم افزار در سرور است تا با ایجاد Clone و یا Imaging در اجرا بتوان به چندین کاربر از یک برنامه خدمات دهی انجام شود . اما برنامه نویسی برای شبکه انجام شده است .

در شبکه های مایکروسافتی با استفاده از فایل msi. میتوان در سرویس WSUS و GPM برای کلاینت ها یک نرم افزار را در چندین پنجره اجرا کرد و بار پردازشی را در سرور پیاده کرد تا سرعت و فضای کلاینت ها افزایش یابد ، این روش اجرای مرکزی نامیده می‌شود .

پایگاه داده : " در مجموع کار این روش ، بیشتر فعالیت و ساختار به دیتا بیس مربوط میشود . یعنی شما با استفاده از زبان های مانند : PHP یا ASP.net یا از این دست زبان تحت وب ، میتوانید برای یک شبکه نرم افزاری طراحی کنید که هم خود نرم افزار روی شبکه اجرا شود هم دیتا بیس در سرور مرکزی مدیریت و حفاظت شود"

فرض کنید برای یک بیمارستان که دارای چندین شاخه و تخصص خاص میباشد و درخواست های روزانه بالا به دلیل مراجعه بیمار دارد ، قرار است یک سیستم اتوماسیون یا ویزیتور طراحی کنید که تمام شبکه بیمارستان در بخش اداری بتواند از آن استفاده کند ، اما محدویت ها و سمت های شغلی مهم هستند و باید چندین سطح دسترسی تعریف کرد .

در اینجا متوجه میشویم که دیتا بیس چقدر حائز اهمیت است و ساخت و ایجاد یک دیتا بیس درست با در نظر گرفتن حداقل افزونگی در اطلاعات چقدر مفید و باعث سرعت نرم افزار میشود .

با تعریف نام کاربری و تعیین محدوده دسترسی کلاینت ها ، میتوان ساخت و ویرایش در نرم افزار توسط هر شخص دارای دسترسی را کنترل کرد .

مثال : " مدیر کل بیمارستان با دسترسی کامل میتواند نظارت جامع بر دیگر کارکنان بیمارستان که با نرم افزار در ارتباط هستن را داشته باشد . اما یک منشی صرفا ویزیتور نمیتواند به نظارت بر امور دیگران در نرم افزار بپردازد " ، همه اینها از اهمیت های ارتباط درست میان دیتا بیس و زبان برنامه نویسی شما میباشد حال چه PHP و چه دیگر زبان های پر کاربرد در این زمینه



نتیجه گیری !

در مجموع مطالب باید متوجه شده باشید که در هر دو صورت این دو روش تحت شبکه و برای شبکه ایجاد میشوند ، پس اینکه شما با علاقه خود تصمیم بگیرید که به کدام سمت میخواهید بروید درست است . زیرا علاقه باعث میشود تا فرد با شور و هیجان بالایی به سمت کار کردن حرکت کند و نتیجه چیزی جز موفقیت نخواهد بود .

" با آرزوی موفقیت روز افزون "

@abdolhayb

بیشتر از یک زبان

شرکت اپل برای به رخ کشیدن قدرت خود در مقابل برند های دیگر، امکانات بسیاری در اختیار توسعه دهندگان برنامه قرار داده است تا بتوانند برنامه های بسیاری را طراحی کنند.

اپل زبان های بسیاری را پشتیبانی می کند و دست برنامه نویسان را برای کد نویسی باز گذاشته است که هر برنامه نویس با زبان مورد علاقه خود برنامه ای برای سیستم عامل هایی مانند MacOS، Watchos، iOS، tvOS و حتی برای سیستم عامل لینوکس بسازند.

یکی از زبان هایی که مورد استفاده توسعه دهندگان قرار دارد ، زبان Swift می باشد ، این زبان قابلیت و امکانات متعددی دارد که در ادامه به آن می پردازیم.



تاریخچه Swift

زبان برنامه نویسی Swift به وسیله Chris Lattner و شرکت اپل در دوم ژوئن ۲۰۱۴ در کنفرانس WWDC معرفی و در دسترس قرار گرفت اما پایه ریزی آن در جولای سال ۲۰۱۰ آغاز شد و این کار به وسیلهChris Lattner و بسیاری از برنامه نویسان شرکت اپل انجام پذیرفت. Swift از زبان های برنامه نویسی نظیرObjective-C Phyton, Ruby, Rust و غیره بوجود امده است و همچنین سازگار با فریم ورک هایی نظیر Cocoa و Cocoa Touch می باشد.

قابلیت های Swift

Swift بهترین جایگزین مناسب برای زبان برنامه نویسی Objective-C می باشد که سعی سازندگان آن بر این بوده است تا با مفاهیم روان و مدرن یک دستور ساده برای نوشتن برنامه استفاده شود و بتوان آن را به روشی آسان توضیح داد. در زبان Swift بر خلاف زبان Objective-C نمایش دادن Pointers و ارزیاب غیرامن به صورت پیشفرض غیرفعال می باشد.

یکی از مهمترین امکاناتی که در Swift قرار داده شده است ، دارا بودن انواع گزینه ها (option type) می باشد. به وسیله این گزینه شما اجازه خواهید داشت تا مقادیر را همانند مقادیری که در زبان C تعریف می کردید مشابه آن تعریف کنید .

یکی از قابلیت های Swift برخوردار بودن از Protocol-oriented programming می باشد ، این به چه معنا می باشد؟ ویژگی ذکر شده یکی از الگوهای OOP می باشد که در این زبان ترجیح داده شده از پروتکل در ساختار کلی برنامه استفاده شود. در کنفرانس WWDC که در سال ۲۰۱۵ برگزار شد با معرفی این ویژگی، نشان داده شد که برای بهتر کردن کدهای برنامه نویسی چگونه می توان از پروتکل ها استفاده کرد.

داشتن کتابخانه قوی ، زمان اجرا و قابل رشد بودن در زبان Swift یکی از نقاط قوت این برنامه می باشد. اگرچه همانند Objective-C ازruntime یکسانی برخوردار می باشد ، ولیکن برای استفاده از این زبان نیاز است تا iOS 7 و یاmacOS نسخه ۱۰،۹ یا بالاتر داشته باشید. همکاری موفق Swiftو Objective-C در یک برنامه باعث شده است تا توسعه و رشد پذیری در برنامه پیشرفت کند ، اما در مقابل نمی توان از زبان های Cو C++ به صورت مستقیم در Swift استفاده کرد. راهی وجود دارد تا از این ابزار در کنار هم بهره برد که برای این کار باید یک پوشش دهنده Objective-Cیا C مابین Swift و C++ ساخته شود.



زبان Swift تاکنون چند نسخه ارائه داده است ، آخرین نسخه آن ورژن ۳ می باشد که در سیزدهم ژوئن ۲۰۱۶ به نمایش در آمد ولی نسخه پایدار آن ۲،۲،۱ می باشد که سوم می ۲۰۱۶ در دسترس عموم قرار گرفت. Swift تحت لیسانس Apache می باشد و از ورژن ۲،۲ به بالا به همین روال می باشد. سیستم عامل های Drawin Linux و FreeBSD محیط اولیه ساخت زبان Swift می باشند.

Swift نسخه ۳

اولین انتشار این زبان برنامه نویسی به صورت Open Sourceمی باشد که با امکاناتی نظیر کد منبع، باگ یاب و mailing list در دسترس همه می باشد. قابلیت های زیاد دیگری به این زبان اضافه شده است که در ذیل به تعدادی از آنها اشاره می کنیم. نام گذاری API به صورت خلاصه شده، مدرن بودن زبان ، طراحی شده به صورت ایمن و داشتن سرعت و قدرت بالا در کامپایل و پردازش. کار با این زبان برنامه نویسی بسیار ساده می باشد و بخصوص در کنار ابزار قدرتمندی مانند Xcode می توان برنامه های بسیار زیبا و شکیل طراحی کرد . علاقمندان و آن دسته از افرادی که دیوایس iPad دارند به راحتی قادر خواهند بود تا swift را درون iPad آموزش ببینند. فقط کافیست برنامه را از app store دانلود کرده تا نمونه و طرح های که برای آموزش تهیه شده است را مشاهده کنند.

@Badrinex

R

آر یک زبان برنامه نویسی متن باز بوده و برای محاسبات آماری و داده ها میباشد که تحت پروانه عمومی همگانی گنو یا به اختصار جی پی ال عرضه شده است و بر اساس زبان های اس و اسکیم توسط رابرت جنتلمن(آماردان) و راس ایحاکا در دانشگاه آکلند در سال ۱۹۹۲ ایجاد شده است.

زبان آر یک زبان مفسری میباشد , یعنی مانند اکثر زبان های مفسری دیگر , کاربر میتواند از طریق خط فرمان مستقیما با مفسر این زبان در تعامل باشد و دستورات را اجرا کند و همچنین این زبان از برنامه نویسی شی گرا یا آبجکت اورنیتید پروگرمینگ هم پشتیبانی می کند.
تحلیلگران آماری, تحلیلگران مالی , دانشمندان داده , آمارداران و آماردانان و هرکسی که نیاز به کار با داده ها و تحلیل آنها دارد از آر استفاده میکنند همینطور افراد زیادی در حوضه اقتصاد از این زبان به عنوان زبان اصلی برنامه نویسی خود استفاده میکنند و بسیازی از ابزار کار خود را با استفاده از آر گسترش میدهند و همچنین متخصصین داده از این زبان برای محاسبات , نمایش و کار بر روی داده های خود و اطلاعات آماری داده های خود استفاده می کنند.

استفاده از زبان آر برای ساخت برنامه های آماری میباشد به طوری که شما با نوشتن اسکریپت ها و توابع داده هارا تحلیل می کنید.

داخل این زبان شی ها , توابع و عملگر هایی وجود دارد که روال تحلیل داده و مدل سازی را راحت میکند.این زبان و کتابخانه هایی که در احتیار دارد توانایی پیاده سازی تعداد وسیع و زیادی تکنیک های آماری و گرافیکی را دارد.

برای وظایف محاسباتی فشرده, کاربران میتوانند کد های سی , سی پلاس پلاس و فورترن را پیوند دهند و در زمان اجرا آنهارا صدا و استفاده کنند و همچنین در سطوح بالاتر , کاربران خیلی پیشرفته می تواند از کد های سی , سی پلاس پلاس , جاوا, پایتون و همچنین چهارچوب دات نت برای دس‌تکاری مستقیم روی اشیاء این زبان استفاده کنند

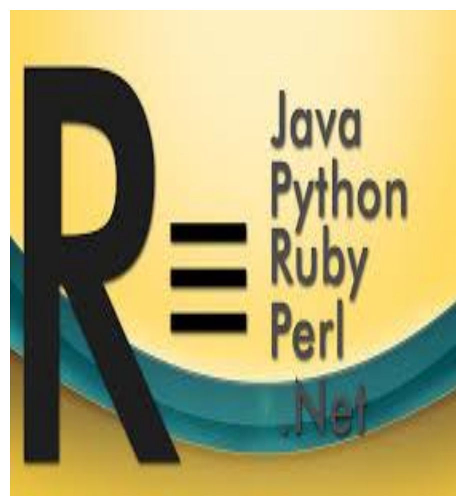
آر از طریق استفاده از پکیج هایی که کاربران این زبان برای آن در زمینه های مختلف آماری و یا ... گسترش میدهند ,این پکیج ها که در درجه اول با آر گسترش داده میشوند و در وهله بعدی با زبان هایی مانند جاوا , سی پلاس پلاس , سی و .فورترن نوشته میشوند,

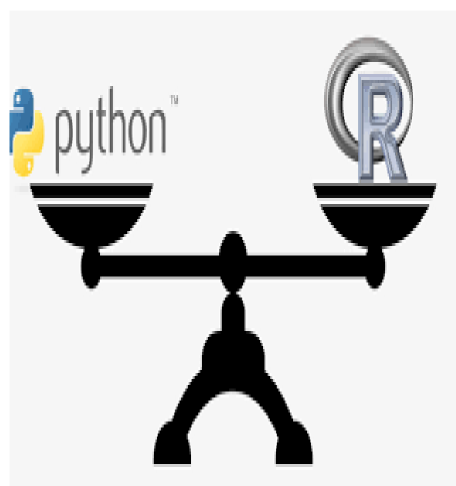
موجب تبدیل شدن آر به یک زبان خیلی گسترش پذیر شده اند, به طوری که توانایی های این زبان تا حد زیادی به واسطه پکیج های آن تشدید شده است.

از جمله کمپانی ها و شرکت های بزرگی که از آر استفاده میکنند میتوان از گوگل ، فیسبوک ، بینگ ، مایکروسافت ، توییتر و موزیلا نام برد.

@TheNonexistent








قبیله گیک ها




قبیله
گیک ها

